

ISSN 1390-6712

MASKAY



ELECTRÓNICA

Vol. 4, No. 1

Análisis del comportamiento interactivo del cliente para diseñar políticas de selección de pares para protocolos basados en BitTorrent*

Marcus V. M. Rocha¹ y Carlo Kleber da S. Rodrigues²

Resumen— Este artículo propone un algoritmo para ser utilizado como base para diseñar políticas de selección de pares de protocolos basados en BitTorrent, teniendo en cuenta aplicaciones multimedia del tipo *on-demand streaming* y los escenarios interactivos. Para ello, se analiza el comportamiento interactivo del cliente al acceder a sistemas multimedia reales. Este análisis consiste en examinar cargas de trabajo de proveedores de contenidos reales y evaluar tres parámetros importantes, a saber, la dispersión temporal, la dispersión espacial, y la popularidad de la posición del objeto multimedia. Estos parámetros se utilizan como las principales pautas para escribir el algoritmo. Por último, la conclusión de este artículo se dibuja con desafíos y posibles trabajos futuros en este campo de investigación.

Palabras clave— BitTorrent; Multimedia; Protocolos.

I. INTRODUCCIÓN

La arquitectura de red *peer-to-peer* (entre pares o entre iguales) ya se ha demostrado ser una solución más eficaz que la arquitectura de red clásica del tipo cliente-servidor para la replicación de contenidos en Internet [1, 2].

Los protocolos para ese tipo de arquitectura de red, protocolos *peer-to-peer* o simplemente protocolos P2P, se basan en la definición de dos políticas fundamentales: la selección de pedazos y la selección de pares (vecinos). La primera se refiere a cómo van los pedazos de un objeto multimedia a ser elegidos por un par (vecino), mientras que la segunda dicta que pares van a recibir los pedazos [2]. Estas dos políticas deben estimular la cooperación y la reciprocidad entre los pares.

La gran eficiencia de los protocolos P2P sin duda ha atraído a una gran cantidad de investigadores de la academia y de la industria, y ha motivado en particular una serie de propuestas para adaptar estos protocolos para satisfacer los requisitos de los sistemas de aplicaciones multimedia,

teniendo en cuenta los escenarios interactivos. En general, estos requisitos se refieren a necesidades de tiempo, a saber: el cliente quiere reproducir el objeto en cuanto se llega al sistema y no tolera ninguna discontinuidad en su transmisión [2, 3, 4].

Bajo este contexto, este artículo propone un algoritmo simple para ser utilizado como base para diseñar políticas de selección de pares de protocolos basados en BitTorrent, teniendo en cuenta aplicaciones multimedia del tipo *on-demand streaming* y los escenarios interactivos. Para ello, se analiza el comportamiento interactivo del cliente al acceder a sistemas multimedia reales.

Ese análisis consiste en examinar cargas de trabajo de proveedores de contenidos reales y evaluar tres parámetros importantes, a saber, la dispersión temporal, la dispersión espacial y la popularidad de la posición del objeto multimedia. Estos parámetros se utilizan como las principales pautas para escribir el algoritmo que se propone. A lo mejor de nuestro conocimiento, esta es la primera vez que se considera el comportamiento interactivo del cliente para ese fin. Por último, la conclusión de este artículo se dibuja con desafíos y posibles trabajos futuros en este campo de investigación.

El resto de este texto está organizado de la siguiente manera. En la Sección II, se explican brevemente el funcionamiento del protocolo BitTorrent y los aspectos y métricas que son considerados para el diseño de sistemas multimedia P2P. La Sección III presenta la caracterización de las cargas de trabajos que usamos en este trabajo y la definición de los parámetros que son empleados en nuestro análisis. En la Sección IV, tenemos los resultados y el análisis de los mismos. La Sección V presenta el algoritmo. Por último, la conclusión y los trabajos futuros se incluyen en la Sección VI.

II. FUNDAMENTOS

A. BitTorrent Protocol

BitTorrent es uno de los protocolos P2P más populares. Los objetos multimedia se dividen en pedazos (típicamente de 256 kB de tamaño) y cada pedazo se divide en bloques (típicamente de 16 kB de tamaño). Rompiendo pedazos en bloques permite que el protocolo mantenga siempre varias solicitudes (generalmente cinco) pendientes a la vez. Cada vez que llega un bloque, se envía una nueva solicitud para un

*Este artículo es una versión resumida del texto publicado en [18].

¹Marcus M. V. Rocha es Doctor en Ciencias por la Universidad de Minas Gerais, en Brazil. Trabaja en la Assembléia del Estado de Minas Gerais, en Brazil (email: marcus@melo-rocha.gov).

²Carlo Kleber da S. Rodrigues es Doctor en Ingeniería de Sistemas y Computación por la Universidad Federal de Rio de Janeiro, en Brazil. Trabaja como Asesor Militar del Ejército de Brazil en Ecuador, Profesor del Centro Universitario de Brasília, en Brazil, y Profesor del Departamento de Eléctrica y Electrónica en la Universidad de las Fuerzas Armadas ESPE, Sangolquí, Ecuador (email: carlokleber@gmail.com).

nuevo bloque. Eso ayuda a evitar el retraso entre los pedazos que se envían, lo que es desastroso para las tasas de transferencia. Los bloques son, por lo tanto, la unidad de transmisión en la red [5, 6].

Pares que todavía están descargando contenidos pueden servir los pedazos que ya tienen a los demás. *Leechers* son pares que sólo tienen algunos o ninguno de los pedazos, mientras que *seeds* son pares que tienen todos los pedazos, pero que permanecen en el sistema para permitir que otros pares puedan descargar desde los mismos. Un conjunto de pares que comparten simultáneamente contenidos de interés común se llama *swarm* [5, 6].

Para participar en el *swarm*, un nuevo par tiene en primer lugar que descargar un archivo estático con la extensión *.torrent* desde un servidor *web* normal. El archivo *.torrent* tiene metadatos (nombre, tamaño, información de *hash*, etc.) del objeto multimedia deseado. Así, el nuevo par es capaz de ponerse en contacto con una entidad centralizada del *swarm* llamada *tracker*. La principal responsabilidad de esta entidad es controlar los participantes del *swarm* y fornecer al nuevo par una lista aleatoria *L* de pares que ya forman parte del *swarm* [5, 6].

El nuevo par intenta entonces establecer conexiones persistentes bidireccionales *TCP* con un conjunto aleatorio de pares (entre 40 y 80) de la lista *L* que acabó de recibir. Los pares que son contactados exitosamente forman el conjunto de vecinos del nuevo par [5, 6].

B. Aspectos y Métricas

En general, tenemos la influencia de los siguientes seis aspectos que afectan sistemas P2P *on-demand* para la transmisión de video [1, 3, 7]: (i) *freeride*, que es el hecho de no compartir la capacidad de carga (*upload*); (ii) ataques maliciosos; (iii) anchos de banda diferentes entre pares; (iv) la presencia de pares no conectables, o sea, que no poseen una dirección globalmente accesible; (v) la coexistencia de pares que hacen video *on-demand* (VoD) con pares que hacen la transferencia de objetos que no son del tipo multimedia; y, por fin, (vi) el comportamiento del cliente.

Conjeturamos que se pueden tener en cuenta esos mismos aspectos al considerar de manera general los sistemas P2P *on-demand*. Esto es bastante razonable debido a que un sistema de video *on-demand* puede ser visto como una clase particular del más general sistema P2P *on-demand*, en el que los objetos son específicamente videos, es decir, una grabación de imágenes en movimiento y con sonido.

Con esto en mente, la pregunta que sigue inmediatamente es cuáles de esos aspectos deben ser entonces considerados cuando se piensa en el diseño de las políticas de selección de pares para los sistemas multimedia P2P *on-demand*. La respuesta es que los aspectos (i), (iii) y (vi) deben ser considerados, ya que son los que intrínsecamente están relacionados con el propio grupo de pares; los demás aspectos son en su mayoría relacionados con el sistema y la infraestructura de la red, seguridad o servicio.

En cuanto a las métricas de desempeño, hay varios de ellos [1, 3, 7]. Sólo para mencionar algunos, tenemos: (i)

índice de continuidad, es decir, la proporción de pedazos recibidos antes del tiempo límite sobre el número total de pedazos; (ii) la demora de inicio, es decir, el tiempo que un cliente tiene que esperar antes de iniciar la reproducción; (iii) el tiempo medio para volver la reproducción después de una interrupción; (iv) el número medio de interrupciones durante la reproducción del objeto multimedia; (v) el tiempo total de descarga del objeto; (vi) el tiempo de arranque, es decir, el tiempo que un cliente tiene que esperar para obtener su primero pedazo; (vii) la utilización del enlace; por fin, (viii) la equidad.

Del mismo modo, la pregunta que sigue es cuáles de las métricas anteriores deben ser considerados para el diseño de políticas de selección de pares de protocolos P2P para sistemas *on-demand*. La respuesta es simple: todas las métricas mencionadas anteriormente pueden ser considerados. Sin embargo, reconocemos la alta complejidad de este pensamiento, ya que los valores resultantes provienen de influencias simultáneas de las dos políticas que constituyen la base para los protocolos P2P: la selección de pares y la selección de pedazos. Por otra parte, es muy intrincada o quizás imposible aislar cada una de estas influencias.

III. CARACTERIZACIÓN DE CARGAS Y PARÁMETROS DE ANÁLISIS

A. Caracterización de las Cargas de Trabajo

Nuestro análisis considera las cargas de trabajo interactivas de tres dominios de aplicación diferentes. Los dominios son audio no académico, video no académico y video académico.

En el ámbito académico, nos fijamos en las cargas de trabajo de Eteach [8], un sistema de video de la Universidad de Wisconsin-Madison, con 46.958 solicitudes a videos de anuncios comerciales de hasta cinco minutos, así como videos educativos de 50 hasta 60 minutos. Además, consideramos un registro de tres años del sistema Manic [9], un sistema de video educativo de la Universidad de Massachusetts, con 25.833 solicitudes.

Para el ámbito no académico, se analizan audio y video, por lo general de menos de 10 minutos, a partir de dos grandes proveedores de contenido de América Latina. El primero es UOL [10], con 5.385.822 solicitudes a objetos de radio y 1.453.117 solicitudes a objetos de video. El segundo, en virtud de la confidencialidad, es denotado simplemente como ISP (Proveedor de Servicios de Internet) y tiene 4.160.889 solicitudes a objetos de radio. Estas cargas también fueron consideradas en las obras de Costa et al. [11] y Rocha et al. [12], pero con el foco en exclusivamente arquitecturas del tipo cliente-servidor.

Para efecto de análisis, agrupamos nuestras cargas de trabajo en tres perfiles de interactividad: alta interactividad (HI – *High Interactivity*): solicitud de corta duración (menos de 20% del tamaño del objeto) y al menos tres solicitudes en una misma sesión, típico de videos académicos; baja interactividad (LI – *Low Interactivity*): solicitud de duración más larga (al menos 20% del tamaño del objeto) y menos de dos solicitudes en una misma sesión, típico de audio no

académicos y videos muy cortos; medio interactividad (MI – *Medium Interactivity*): duración de solicitud más corta (menos de 20% del tamaño del objeto multimedia), menos de tres solicitudes en una misma sesión, típico de vídeos no académicos.

En total, contamos con 40 cargas de trabajo distintas, las cuales se distribuyen de la siguiente manera: 19 cargas de trabajo de HI; 8 cargas de trabajo de MI, y 13 cargas de trabajo de LI. Por otra parte, teniendo en cuenta el dominio de aplicación, se distribuyen de la siguiente manera: 8 cargas de trabajo comerciales de audio, 10 cargas de trabajo de video comerciales, y 22 cargas de trabajo de videos educativos.

B. Parámetros de Análisis

Conjeturamos que podemos decidir de manera eficiente sobre la selección de pares (vecinos) si podemos evaluar los siguientes valores: (1) Cuanto las solicitudes divergen entre sí teniendo en cuenta los tiempos de llegada; (2) Cuanto de los pedazos recuperados pueden ser efectivamente compartidos; (3) Cual es la frecuencia que se pide a cada posición del objeto. Para evaluar estos tres valores, se definen tres parámetros diferentes: dispersión temporal, dispersión espacial, y la popularidad de la posición del objeto.

Dispersión temporal es sólo otra manera para denotar la tasa de solicitud y se puede utilizarla para calcular el valor (1). Por su parte, la dispersión espacial se refiere a la cantidad de contenido de las solicitudes que dos clientes distintos consecutivos tienen en común. Se utiliza para calcular el valor (2). Si las solicitudes tienen un contenido en común, o sea, una superposición, este contenido puede ser potencialmente compartido. Si esta superposición aumenta, entonces la dispersión espacial disminuye. Por otro lado, si esta superposición disminuye, la dispersión espacial aumenta.

Evaluar la dispersión temporal es sencillo, ya que es inversamente proporcional a la tasa de solicitud. Por su parte, para medirse la dispersión espacial se procede de la siguiente manera. Sea Q_p el número de veces que una posición del objeto p es solicitada por los clientes del sistema durante todo el tiempo de reproducción. Por lo tanto, Q_p es la popularidad de la posición p del objeto. Cualquier posición p , recibida por un par, está potencialmente disponible para ser compartida con otros pares. De ello, se desprende que se define el potencial para el intercambio de contenidos P como sigue.

$$P = \sum_{p=0}^{T-1} (Q_p - 1) \quad (1)$$

Ahora, para medir la dispersión espacial D , se compara la cantidad de contenido que se recupera M con el potencial para el intercambio de contenidos P como sigue.

$$D = 1 - \left(\frac{P}{M} \right) \quad (2)$$

Se puede notar que $P < M$, y que D se encuentra en el intervalo $[0, 1)$. Además de eso, Q_p se utiliza para cuantificar el valor (3) y es simple para ser calculado. Como ya se ha dicho, todo lo que tenemos que hacer es contar el número de veces que se pidió la posición p del objeto.

Para facilitar la discusión de los resultados, consideramos los siguientes intervalos para una categorización numérica de la dispersión: menor que 0,1 (baja dispersión espacial); dentro del intervalo $(0,1; 0,5]$ (media dispersión espacial); mayor que 0,5 (alta dispersión espacial).

Por fin, para concluir esta sección, es importante decir que la dispersión espacial, dispersión temporal, y la popularidad de la posición del objeto son especialmente capaces de captar de manera conjunta las influencias procedentes de los aspectos y métricas que explicamos en la Sección II, lo que hace que el análisis que deseamos sea mucho más simple y, por lo tanto, sin duda más viable.

IV. RESULTADOS Y ANÁLISIS

A. Dispersión Temporal

La mayoría de nuestras cargas de trabajo presentan una alta dispersión temporal. Esto se explica por el hecho de que, en los servidores multimedia reales, los clientes no vienen en multitudes, ya que individualmente tienen diferentes necesidades y diferentes disponibilidad de tiempo para acceder a los contenidos disponibles.

Por ejemplo, en los sistemas académicos, los contenidos que se acceden se refieren principalmente a las clases de video que se pueden ver en varios momentos diferentes durante el día y la noche, ya que los estudiantes no se limitan a horarios específicos en absoluto. Por otra parte, teniendo en cuenta los sistemas P2P analizados en [13], se sabe que alrededor de 40% a 70% de los *swarms* tienen sólo tres o menos pares, y que más de 70% de los *swarms* son de tamaño más pequeño que diez [14]. Estos números refuerzan la creencia general de que la mayoría de los escenarios presentan cargas de trabajo con alta dispersión temporal.

Por lo tanto, en lugar de tener una evaluación cuantitativa de la dispersión temporal, simplemente consideramos el siguiente pensamiento: como la baja dispersión temporal puede, hasta cierto punto, ayudar a compartir contenidos, en virtud de su eventual impacto sobre la dispersión espacial, se debe entonces seleccionar como vecinos los pares con la dispersión temporal más baja posible.

B. Dispersión Espacial

La Fig. 1 tiene las cargas de trabajo en función de los perfiles de interactividad y el objetivo es examinar cómo la dispersión espacial varía para cada uno de estos perfiles. El eje y tiene la dispersión de cada carga de trabajo. El eje x muestra un índice arbitrario de la carga de trabajo dentro de su perfil de interactividad. Se puede percibir que los perfiles

de LI proporcionan valores muy bajos de dispersión, los perfiles MI presentan valores intermedios de dispersión, y los perfiles de HI tienen altos valores de dispersión.

La Fig. 2 presenta las cargas de trabajo agrupadas por el dominio de la aplicación y el objetivo es evaluar cómo la dispersión espacial varía dentro de cada uno de estos dominios. El eje y trae la dispersión de cada carga de trabajo. El eje x muestra un índice arbitrario de la carga de trabajo dentro de su dominio de aplicación: vídeo académico, vídeo comercial o audio comercial. Aunque la dispersión pueda variar dentro de un mismo dominio, se puede notar una tendencia que nos permite conjeturar que la dispersión es alta para los videos educativos, media para los vídeos comerciales, y baja para el audio comercial.

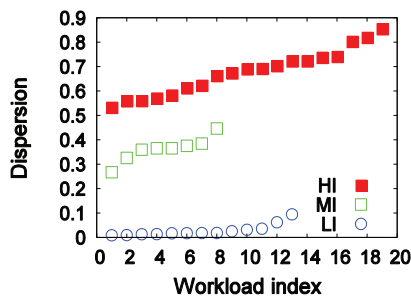


Fig 1. Dispersión espacial (perfil de interactividad).

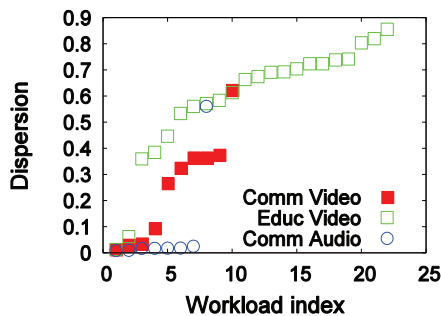


Fig 2. Dispersión espacial (dominio de aplicación).

Las Figuras 3, 4 y 5 también agrupan las cargas de trabajo por el dominio de aplicación, pero esta vez el objetivo es ver cómo la dispersión espacial varía dentro de un mismo dominio y teniendo en cuenta distintos perfiles de interactividad. El eje y tiene la dispersión de cada carga de trabajo, y el eje x un índice arbitrario de la carga de trabajo. Más precisamente, la Fig. 3 muestra que la mayoría de las cargas de trabajo de audio comerciales son del perfil de interactividad baja y de baja dispersión. En la Fig. 4, tenemos que la mayoría de las cargas de trabajo son de perfil de alta interactividad y de media y alta dispersión. Por último, a partir de la Fig. 5, vemos que las cargas de trabajo son principalmente de perfil de interactividad baja y media. Se puede notar que, para el perfil de baja interactividad, tenemos

una baja dispersión y, para el perfil de interactividad media, tenemos media dispersión.

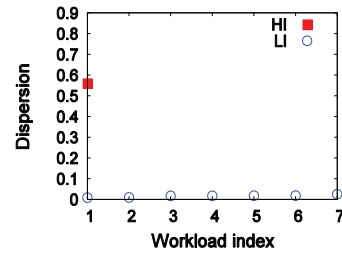


Fig 3. Dispersión espacial (audio comercial).

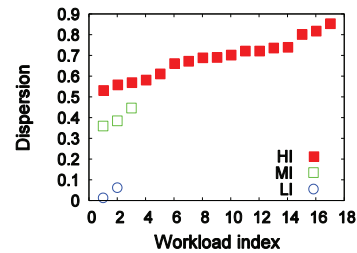


Fig 4. Dispersión espacial (video educativo).

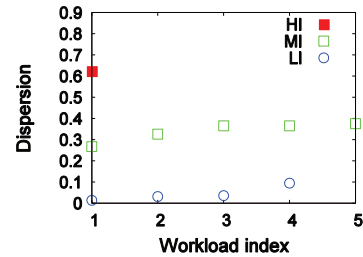


Fig 5. Dispersión espacial (video comercial).

Las Figuras 6, 7 y 8 ilustran la dispersión espacial para las cargas de trabajo agrupadas en función del tamaño del objeto. El objetivo ahora es ver cómo este tipo de dispersión varía en función de este tamaño dentro de los dominios de aplicación. El eje y muestra la dispersión de cada carga de trabajo y el eje x tiene un índice arbitrario de la carga de trabajo. Más precisamente, la Fig. 5 muestra claramente que la mayor parte del contenido de menos de cinco minutos tiene baja dispersión. En otras palabras, no importa el dominio de la aplicación, es decir, los clientes tienden a recuperar objetos más cortos por completo. La Fig. 6 muestra la dispersión de objetos de 5 a 20 minutos. La mayoría de los objetos que caen dentro de esta categoría son del dominio video educativo y muestran una alta dispersión. Finalmente, la Fig. 7 presenta dispersión para los objetos de tamaño mayor que 20 minutos. En este caso, tenemos que la dispersión varía de media a alta.

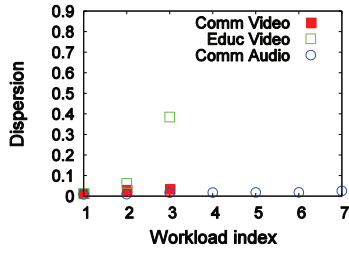


Fig 6. Dispersión espacial (< 5 minutos).

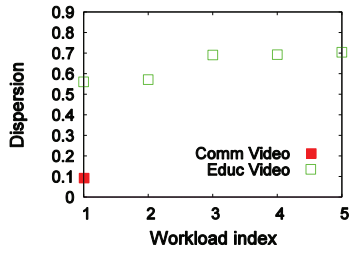


Fig 7. Dispersión espacial (entre 5 y 20 minutos).

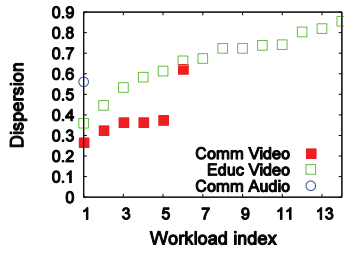


Fig 8. Dispersión espacial (> 20 minutos).

A partir de los resultados observados, se puede concluir que: (i) si la interactividad aumenta, entonces la dispersión también aumenta; (ii) el perfil de la interactividad tiene más influencia sobre el valor de dispersión que el dominio de aplicación; (iii) si el tamaño del objeto aumenta, entonces la dispersión también aumenta.

Por lo tanto, teniendo en cuenta el parámetro dispersión espacial, debemos considerar el siguiente pensamiento en general: *como la baja dispersión espacial ayuda a producir más superposición de contenidos, entonces se debe elegir como vecinos los pares con la dispersión espacial más baja posible.*

C. Popularidad de la Posición del Objeto

La popularidad de la posición del objeto p , denotada por Q_p , tal como se define en la Sección III, es el número de veces que una posición p es solicitada por los pares, teniendo en cuenta todo el tiempo de reproducción. Las Figuras 9, 10 y 11 muestran la popularidad de las posiciones del objeto considerando tres cargas de trabajo y los tres perfiles de interactividad. El objetivo es evaluar cómo el perfil de interactividad influye en la popularidad de la posición. El eje

x tiene cada posición del objeto en segundos. El eje y tiene el número de veces que cada posición del objeto es solicitada.

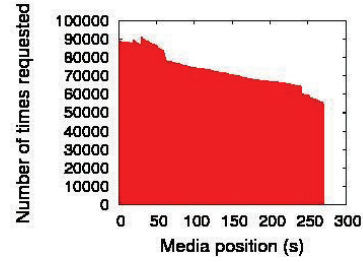


Fig 9. Popularidad (LI).

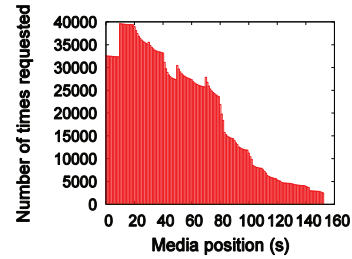


Fig 10. Popularidad (MI).

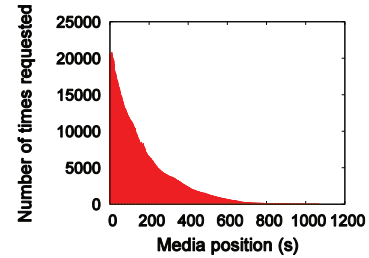


Fig 11. Popularidad (HI).

A partir de los resultados, y considerando todos los perfiles de interactividad, las posiciones más populares se encuentran a menudo al inicio del objeto. Además, si la carga de trabajo es más interactiva, entonces la distribución de popularidad es menos uniforme.

Por lo tanto, teniendo en cuenta la popularidad de la posición del objeto, obtenemos la siguiente reflexión general: *se debe seleccionar como vecinos los pares que ya han recuperado la mayor parte de las primeras posiciones de los objetos, ya que esto resultará en más oportunidades para compartir contenidos.*

V. ALGORITMO

A partir de los resultados y conclusiones de la última sección, el diseño de las políticas de selección de pares debe dictar que los pares que se seleccionarán como vecinos deben ayudar a reducir la dispersión (temporal y espacial) y tener en cuenta la popularidad de la posición del objeto, lo que proporcionará más oportunidades para compartir contenido

y, por lo tanto, la optimización del propio sistema P2P. Con este fin, se propone observar los puntos que siguen.

- 1) Conforme en el protocolo BitTorrent tradicional, un par recién llegado recibe una lista de posibles vecinos del *tracker*;
- 2) Todos los contenidos recuperados se almacenan en búferes locales. Así no hay necesidad de un par solicitar el mismo contenido más de una vez;
- 3) Cada par mantiene un registro del número de veces que se solicita cada posición del objeto, es decir, la popularidad de la posición. Este registro de popularidad se puede utilizar como base para evaluar el potencial para el intercambio de contenidos P en la Ecuación 1 y, por lo tanto, para evaluar la dispersión en la Ecuación 2. Además de este registro, cada par también mantiene un registro de los contenidos del búfer, conforme en el protocolo BitTorrent tradicional;
- 4) Después de recibir la lista de posibles vecinos, el par intercambia su registro de popularidad con los posibles vecinos. El registro de los contenidos del búfer también se intercambia;
- 5) El par puede entonces evaluar la influencia de cada vecino con respecto a la dispersión. De esta manera, un par puede seleccionar a sus vecinos y obtener la mínima dispersión posible, teniendo en cuenta los siguientes pasos.

Paso 1: Sea C el conjunto de pares recibidos del *tracker* y S el conjunto de vecinos seleccionados. Tenga en cuenta que S se configura inicialmente vacío;

Paso 2: Para cada par c de C , evaluar la dispersión de la carga de trabajo considerando el conjunto de pares que tiene c y todos los pares que ya están en S , de la siguiente manera:

Paso 2.1: El par c que resulta la dispersión de la carga de trabajo más baja se selecciona como un nuevo vecino;

Paso 2.2: Este par recientemente seleccionado se elimina de C y se incluye en S ;

Paso 2.3: Detener cuando: C está agotado o S ya cuenta con el número de vecinos indicado por el protocolo BitTorrent original.

Por último, como directrices complementarias para el algoritmo anterior, tenemos los puntos que siguen.

- a) Para las cargas de trabajo LI, donde la mayoría de los clientes recuperan el objeto completo, es preferible seleccionar como vecinos los pares que ya han comenzado a recibir los datos, ya que es más probable que los contenidos buscados estarán disponibles;
- b) Es preferible elegir como vecinos los pares con las tasas más altas de solicitud, es decir, con la dispersión temporal más baja. Pares con tasas de solicitud mayores son más propensos a llenar sus búferes más rápidamente y, por lo

tanto, tienen mayor probabilidad de tener más contenidos para compartir;

- c) Después de formar el grupo de vecinos como se ha descrito anteriormente, las capacidades de carga y descarga del sistema deben ser evaluadas. En caso de que estos valores no sean satisfactorios, el conjunto de vecinos tiene que ser evaluado de nuevo; además de considerar la dispersión (temporal y espacial) y la popularidad, el concepto de reciprocidad indirecta [15] también se debe implementar en la selección del nuevo vecino en el Paso 2.1.

VI. CONCLUSIONES Y TRABAJOS FUTUROS

En este artículo se introdujo un algoritmo para ser utilizado como base para el diseño de las políticas de selección de pares de protocolos basados en el BitTorrent, teniendo en cuenta el comportamiento interactivo del cliente al acceder a sistemas multimedia reales.

Entre los resultados más importantes obtenidos en este trabajo, podemos destacar los siguientes: (i) la mayoría de las cargas de trabajo de los sistemas multimedia analizados presentan una alta dispersión temporal. Esto se debe a que los clientes no vienen en multitudes, ya que tienen individualmente diferentes necesidades y diferente disponibilidad de tiempo para acceder a los contenidos disponibles. Sin embargo, ya que un par usando el protocolo BitTorrent almacena todo el contenido recuperado en búferes locales, la alta dispersión temporal tiene poco impacto sobre la posibilidad de compartirse contenido; (ii) la dispersión espacial es una función directa del nivel de la interactividad del cliente y del tamaño del objeto, y sufre poca influencia del dominio de aplicación; (iii) las posiciones más populares a menudo se encuentran al inicio del objeto, y si la carga de trabajo es más interactiva, entonces la distribución de popularidad es menos uniforme; (iv) el algoritmo propuesto es bastante completo en el sentido de que es capaz de tener en cuenta los tres parámetros definidos en este trabajo, además del concepto de reciprocidad indirecta.

Por último, creemos que para trabajos futuros se pueden incluir especialmente: (i) cuantificar la optimización proporcionada por los puntos de vista de diseño revelados en este trabajo por medio de simulaciones y también despliegues reales, (ii) estudiar la transmisión en vivo en Internet para evaluar las eventuales modificaciones o adaptaciones que deben ocurrir en las políticas de selección de pares y de pedazos y, por último, (iii) analizar la influencia de la personalización y de la adaptación al perfil de capacidad del cliente, así como del servicio de traducción de direcciones (NAT – *Network Address Translation*) en el diseño de protocolos P2P [16, 17].

REFERENCES

- [1] D'Acunto, L., Chiluka, N., Vinkó, T. & Sips, H. "BitTorrent-like P2P approaches for VoD: A comparative study", *Computer Networks*, vol. 57, No. 5, pp 1253 – 1276, 2013.
- [2] Ramzan, N., Park, H. & Izquierdo, E. "Video streaming over P2P networks: Challenges and opportunities", *Signal processing: Image Communication*, Vol. 27, pp 401 – 411, 2012.
- [3] Hoffmann, L. J., Rodrigues, C.K.S. & Leão, R. M. M. "BitTorrent-like protocols for interactive access to VoD systems", *European Journal of Scientific Research*, vol. 58, No. 4, pp 550-569, 2011.
- [4] Varvello, M., Steiner, M. & Laevens, K. "Understanding BitTorrent: a reality check from the ISP's perspective", *Computer Networks*, vol.56, No. 40, pp 1054 – 1065, 2012.
- [5] Cohen, B. "Incentives build robustness in BitTorrent", *First Workshop on Economics of Peer-to-Peer Systems*, Berkeley, EUA, 2003.
- [6] Legout, A., Urvoy-Keller, G. & Michiardi, P. "Rarest first and choke algorithms are enough", *6th ACM SIGCOM Conference on Internet Measurement*, Rio de Janeiro, Brazil, 2006.
- [7] Bharambe, A., Herley, C. & Padmanabhan, V. "Analyzing and improving a BitTorrent network's performance mechanisms", *25th IEEE International Conference on Computer Communications*, Barcelona, Catalunya, Spain, 2006.
- [8] <http://eteach.cs.wisc.edu/index.html>
- [9] <http://manic.cs.umass.edu>.
- [10] <http://www.uol.com.br>.
- [11] Costa, C., Cunha, I., Borges, A., Ramos, C., Rocha, M., Almeida, J., & Ribeiro-Neto, B. "Analyzing Client Interactive Behavior on Streaming Media Servers", *13th WWW Conf.*, New York, USA, 2004.
- [12] Rocha, M., Maia, M., Cunha, I., Almeida, J. & Campos, S. "Scalable Media Streaming to Interactive Users", *ACM MULTIMEDIA*, Singapore, Singapore, 2005.
- [13] Hoßfeld, T., Lehrieder, F., Hock, D., Oechsner, S., Despotovic, Z., Kellerer, W. & Michel, M. "Characterization of BitTorrent swarms and their distribution in the Internet", *Computer Networks*, vol.55, No. 5, pp 1197 – 1215, 2011.
- [14] De Souza e Silva, E., Leão, R., Menasché, D. & Rocha, A. "On the interplay between content popularity and performance in P2P systems", *10th International Conference, QEST 2013*, Buenos Aires, Argentina, 2013.
- [15] Menasché, D., Massoulié, L. & Towsley, D. "Reciprocity and barter in peer-to-peer systems", *29th Conference on Information Communications*, San Diego, CA, USA, 2010.
- [16] Zerkouk, M., Mhamed, A. & Messabih, B. "A user profile based access control model and architecture", *International Journal of Computer Networks & Communications (IJCNC)*, vol. 5, No. 1, pp 171 – 181, 2013.
- [17] Masoud, M. Z. M "Analytical modelling of localized P2P streaming systems under NAT consideration", *International Journal of Computer Networks & Communications (IJCNC)*, vol. 5, No. 3, pp 73 – 89, 2013.
- [18] Rocha, M. V. M. & Rodrigues, C. K. S. "On client's interactive behaviour to design peer selection policies for BitTorrent-like protocols", *International Journal of Computer Networks & Communications (IJCNC)*, vol. 5, No. 5, pp 141-159, 2013.